

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## HLOUBKOVÁ MAPA Z DVOJICE OBRAZŮ

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

FILIP RAAB

BRNO 2013



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**  
**ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ**

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## **HLOUBKOVÁ MAPA Z DVOJICE OBRAZŮ**

DEPTH MAP FROM A PAIR OF IMAGES

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**FILIP RAAB**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. BRONISLAV PŘIBYL**

BRNO 2013

## Abstrakt

Tato práce popisuje proces vytváření hloubkové mapy. K vytváření využívá mapy disparity. V teoretické části jsou popsány základní metody vytváření mapy disparit a její přepočtení na hloubkovou mapu. Je učiněno rozhodnutí výběru implementované metody, která je podrobněji popsána. V praktické části se popisuje implementace zvolené metody a její testování. V závěru jsou shrnuta fakta o metodě, implementaci a možném pokračování projektu.

## Abstract

This thesis describes the process of creating a depth map. It is based on creating disparity map. In theoretical chapters are described basic methods creating disparity map and its relationship to depth map. Here is made decision about choice implemented method, which is in detail described. In practical chapters is description of process creating followed by testing of results. At the end is summary all facts about implemented method, implementation and possible continuation of project.

## Klíčová slova

Hloubková mapa. Mapa disparit. Loopy belief propagation, CSAD. CGRAD.

## Keywords

Depth map. Disparity map. Loopy belief propagation. CSAD. CGRAD

## Citace

Filip Raab: Hloubková mapa z dvojice obrazů, bakalářská práce, Brno, FIT VUT v Brně, 2013

# Hloubková mapa z dvojice obrazů

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Bronislava Příbyla

.....

Filip Raab  
15. května 2013

## Poděkování

Rád bych poděkoval mému vedoucímu Ing. Bronislavu Příbylovi za odborné vedení, trpělivost po celou dobu vytváření této práce a ochotu při častých konzultacích.

© Filip Raab, 2013.

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.*

# Obsah

<b>1 Úvod</b>	<b>2</b>
<b>2 Zpracování obrazu pro odhad</b>	<b>3</b>
2.1 Stereovidění . . . . .	3
2.2 Epipolární geometrie . . . . .	4
2.3 Segmentace . . . . .	4
<b>3 Výpočet disparity</b>	<b>7</b>
3.1 Globální metody . . . . .	8
3.2 Lokální metody . . . . .	9
<b>4 Analýza metod pro výpočet disparity</b>	<b>11</b>
<b>5 Metoda založená na segmentaci a optimalizaci</b>	<b>12</b>
5.1 Barevná segmentace . . . . .	12
5.2 Hledání korespondencí . . . . .	13
5.3 Odhad disparitních rovin . . . . .	13
5.4 Optimalizace . . . . .	14
5.5 Loopy belief propagation . . . . .	15
<b>6 Implementace</b>	<b>17</b>
6.1 Barevná segmentace . . . . .	17
6.2 Hledání korespondencí . . . . .	17
6.3 Odhad disparitních rovin . . . . .	18
6.4 Optimalizace . . . . .	19
6.5 Loopy belief propagation . . . . .	19
6.6 Výpočet hloubky . . . . .	19
<b>7 Testování a diskuze</b>	<b>21</b>
<b>8 Závěr</b>	<b>27</b>
<b>A Disparitní mapy</b>	<b>30</b>
<b>B Hloubkové mapy</b>	<b>33</b>
<b>C Tabulka přesnosti měření</b>	<b>35</b>

# Kapitola 1

## Úvod

Počítače jsou využívány ve všech činnostech naší společnosti. Důvodem je stále se snižující cena a rostoucí výkon. Počítače našly svoje uplatnění i v pracovním životě. Dnes již nikoho nepřekvapí jejich použití v nemocnicích, bankách, ale i v turistice a řemeslné tvorbě. Za zvyšující se oblibu v posledních dvou odvětvích může nemalou měrou obor počítačového vidění. Velká část řemeslných firem dnes vytváří svoje portfolia a pracovní návrhy v některém z programů *CAD*, počítačem podporované projektování. A každá úspěšná cestovní kancelář nabízí virtuální prohlídky svých nejoblíbenějších destinací.

Tato bakalářská práce popisuje z oboru počítačového vidění vytváření hloubkových map, které spadá do oblasti stereovidění. Hloubková mapa zaznamenává vzdálenosti objektů v obraze od objektivu fotoaparátu. Někdy se označuje jako získávání třetího rozměru. Dnes tento obor zaznamenává velký zájem. Avšak principy metod, které jsou využívány pro tuto tvorbu, nejsou žádnou novinkou. Větší část z nich zanedlouho oslaví svoje třicáté narozeniny. Jedná se tedy o metody, které za svou existenci byly několikrát testovány a upravovány. Mohlo by se říci, že jde o odvětví, ve kterém bylo vše již popsáno. To však je jen částečnou pravdou, principy tvorby sice byly již popsány a doladěny, avšak zatím nikdo nedefinoval jejich nejlepší kombinaci. A dnes je stále publikováno nemalé množství nových metod zabývajících se touto problematikou.

Mluví-li se o tvorbě hloubkové mapy, vždy se musí zmínit i disparitní mapa. Zatím se může disparitní mapa považovat za předzpracování dvojice snímků, které měří odlišnosti v různých pohledech při pořizování obrazů. Většina metod využívá této disparitní mapy, ze které až následně získá hloubkovou mapu. Podrobnější její popis bude popsán v kapitole 2, kde bude popsán vztah mezi těmito dvěma mapami. Kapitola 3 bude zaměřena na studium existujících základních metod získávání disparitních map, kde budou obecné metody rozřazeny a popsány. Po studiu metod se udělá průzkum nových pokročilých metod, které využívají principů základních metod, kapitola 4. Zde bude učiněno rozhodnutí pro konkrétní metodu, která v kapitole 5 se blíže specifikuje. Kapitola 6 uvádí rozdíly a poznatky od specifikace, popřípadě postupy, které nebyly ve specifikaci uvedeny. V předposlední kapitole 7 se budou testovat dosažené výsledky disparitních map s referenčními a ukáže se funkčnost tvorby hloubkové mapy na vlastních obrazech. Poslední kapitola 8 bude shrnutí dosažených výsledků, postupů a návrhů na pokračování práce.

## Kapitola 2

# Zpracování obrazu pro odhad

Tato kapitola bude zaměřena na metody spojené s oborem stereovidění. Budou zde popisovány sice i metody, které se využívají v jiných částech počítačového vidění anebo i jinde. Tato jejich vlastnost však bude opomíjena a zaměření bude pouze na ty vlastnosti, které jsou pro toto téma užitečné.

### 2.1 Stereovidění

Stereovidění je obor zabývající se rekonstrukcí třetího rozměru, neboli hloubky [5]. Získávání tohoto rozměru se provádí z dvou a více dvojdimenzionálních obrazů. Metody pracují na principu jako lidské vnímání skrze dvě oči. Tedy obrazy se tvoří dvojicí kamer s rovnoběžnými optickými osami a s dostatečným odstupem objektivů od sebe. Různá poloha objektivů zaznamenává scénu z jiného úhlu. V závislosti na vzdálenosti objektů scény od objektivu, se ve snímku objevují, mizí nebo se posouvají detaily, ale i celé objekty scény. Tyto rozdíly snímky od sebe odlišují. Velikost změn je nepřímo úměrná vzdálenosti objektu od objektivu či oka. Zatímco u objektů vzdálených přes 100 metrů změny v poloze nebo úhlu pohledu lze sotva zaznamenat, u objektů ve vzdálenosti do 30 cm se změny projevují velice znatelně. To je možné vyzkoušet i na této práci, když se bude na tento text dívat střídavě jedním a druhým okem.

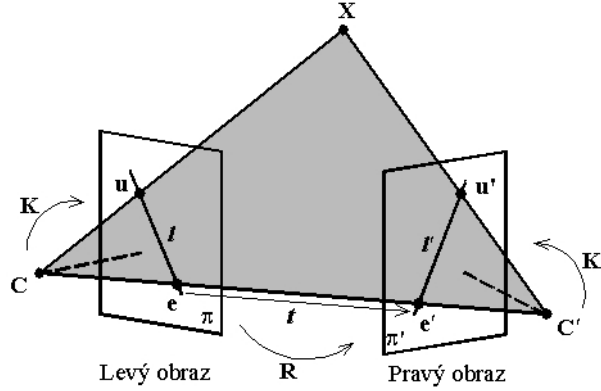
Nyní již je známo, co je náplní oboru stereovidění. Obor stereovidění má i svůj vlastní slovník pojmů. Jakmile bude třeba, pojem bude vysvětlen. Pro pochopení dalších kapitol, týkajících se epipolární geometrie, je třeba vysvětlit první množinou pojmů. Jsou to *disparita*, *disparitní mapa* a *hloubková mapa*.

**Disparita:** Jedná se o vzdálenost mezi souřadnicemi bodů v levém obraze a pravém obraze, které zobrazují ten samý bod scény.

**Disparitní mapa:** Pro každý bod  $(x, y)$  z levého obrazu se určí disparita k bodu v pravém obraze. Tato informace se uloží do disparitní mapy na pozici  $(x, y)$ .

**Hloubková mapa:** Jak název napovídá, jde o obraz, kde každý bod scény nese informaci o jeho hloubce, neboli vzdálenost od objektivu. Tato hodnota je nepřímo úměrná disparitě v disparitní mapě.

## 2.2 Epipolární geometrie



Obrázek 2.1: Epipolární geometrie dvou kamer

Vztah mezi hloubkou a disparitou lze vyjádřit za pomoci *epipolární geometrie*. Epipolární geometrie je geometrie mezi dvěma a více pohledy téže scény. Epipolární geometrie definuje epipolární rovinu. Ta se skládá ze tří bodů  $C, C', X$ .  $C$  a  $C'$  jsou optické středy kamer,  $X$  je sledovaný bod. Protínáním obrazových rovin  $\pi$  a  $\pi'$  epipolární rovinou se získává epipoláry  $l$  a  $l'$ . Známe-li uspořádání kamer, lze podle průsečíků na epipolárách  $l$  a  $l'$  určit polohu bodu  $X$ . Celý postup lze však zjednodušit při dodržení tří pravidel při pořizování obrazů. Za předpokladu, že roviny obrazů budou při pořizování v rovině optické osy rovnoběžné, kamery se dívají stejným směrem, kamery mají stejné nastavení ohniskové vzdálenosti, lze celý proces získávání hloubky zjednodušit na vztah mezi disparitou bodu a hloubkou bodu. To lze vyjádřit rovnicí

$$h = \frac{f}{d} \cdot |C - C'|, \quad (2.1)$$

kde  $h$  je hloubka,  $d$  je disparita,  $f$  je ohnisková vzdálenost a rozdíl  $|C - C'|$  udává vzdálenost objektivů od sebe.

Vzdálenost objektivů od sebe a ohniskovou vzdálenost lze zaznamenat při pořizování snímků. Zbývá tedy získání disparity.

Proto, aby bylo možno vypočítat disparitu všech bodů, disparitní mapu, je třeba znát, které body v levém obraze a v pravém obraze si odpovídají. Operaci se říká hledání *korespondencí*, tj. dvojice obrazových bodů  $(P_p, P_l)$ , které odpovídají stejnému bodu scény. Způsobů jak se takové korespondence hledají, je několik. Obecné způsoby jsou popsány v kapitole 3.

Po získání disparitní mapy stačí pak každý bod přepočítat podle rovnice hloubky 2.1.

## 2.3 Segmentace

Segmentace obrazu odpovídá rozdělení obrazů do množiny oblastí, *segmentů*, tj. stejno-barevných oblastí, které následně celý obraz překryjí. Cílem je, aby segmenty odpovídaly reálným částem scény [11].



Obecný důvod proč se segmentuje [11] je, že se jedná o předzpracování obrazu pro další úlohy.

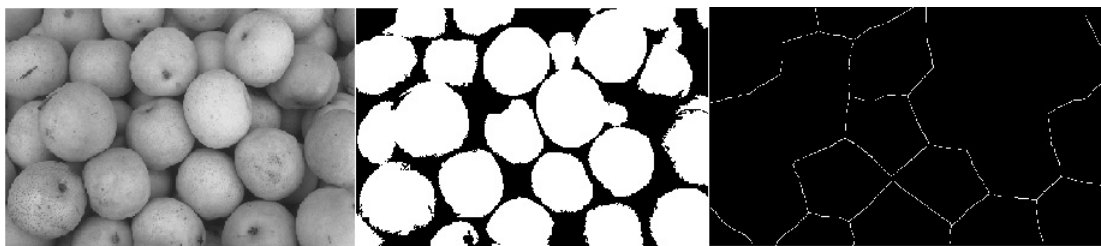
Těmi můžou být změny v zobrazení, organizace bodů obrazu do vyšší scény, kde se body shlukují do segmentů, někdy se označují jako *superpixels*.

## Iterativní K-means shlukování

Jde o iterativní algoritmus podobající se horolezeckým metodám (ang. hill-climbing), metoda hledání optimální cesty v grafu. Pro vysvětlení způsobu práce algoritmu K-means je nejjednodušší popis algoritmem.

1. Nastaví se počítadlo iterací  $ic$  na 1
2. Náhodně se zvolí středy segmentů (ang. K means clusters, kde K je počet klastrů)
3. Pro každý vektor  $x_i$  se spočítá jeho vzdálenost do středu segmentu. Těchto výpočtů bude tolik, kolik je středů segmentů.  $x_i$  je přiřazeno do segmentu  $C_j$ , ke kterému má nejbližší.
4. Nyní se inkrementuje počítadlo a znova se přepočítají středy segmentu, většinou průměrnou hodnotou.
5. Opakuje se krok 3 a 4 dokud  $C_k(ic) = C_k(ic + 1)$  pro všechny segmenty, tj. žádný bod nezměnil segment.

## Watershad



Obrázek 2.2: watershad: původní obraz, simulovaná vodní hladina oddělující segmenty, minimalizace hranic a slučování segmentů

Práci algoritmu lze popsat jako zaplavování krajiny vodou viz 2.2. Jednotlivá zvětšující se „jezírka“ reprezentují segmenty, rozvodí mezi nimi vytváří rozdělovací linie kolem objektů. V šedotónovém obrazu odstín šedi odpovídá úrovni, kam až hladina dosáhne. Algoritmus obsahuje problém přesegmentování v případě, že v obraze je hodně *šumu*, náhodné rušivé hodnoty. Problém se dá řešit výpočtem gradientu a pak až použít watershed algoritmu a nebo užitím efektivního algoritmu po watershad, ten eliminuje ty části, které mají vysoký gradient.

## Barevná segmentace

V této práci tento druh segmentace bude popsán o něco lépe, jde totiž o segmentaci, která je zde použita.

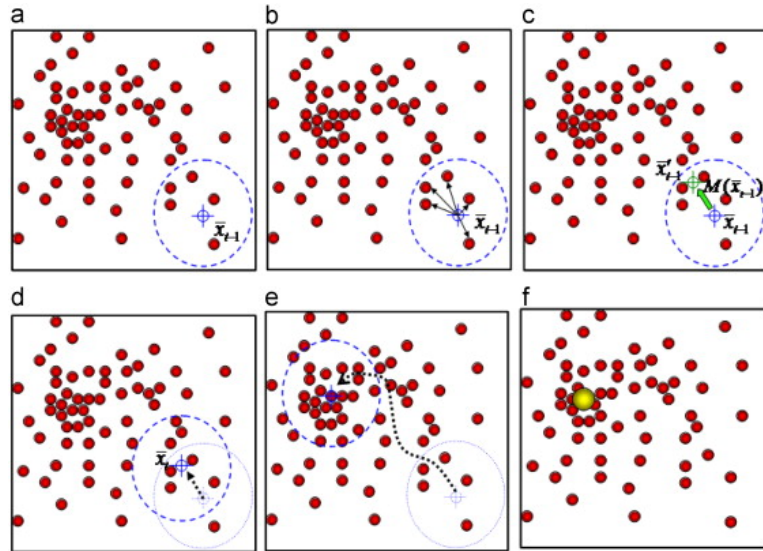
Tato segmentace je popsána článkem [3], nejedná se pouze o tvorbu segmentací. Jde také o tvorbu disparitní mapy využívající barevné segmentace na principu Mean-shift. To byl nejspíš důvod, aby byla tato segmentace použita v práci [6], která je použita zde pro implementaci výpočtu disparitní mapy.

Předpokladem na kterém je metoda postavena je, že disparita se bude v segmentu měnit plynule. A změny se mohou vyskytnout pouze na okrajích těchto segmentů, které nejvěrohodněji zachycují strukturu scény.

## Mean-shift

V obvyklém postupu vytváření klastrů, jsou příznaky modelovány jako směs vícerozměrných normálních rozdělání (ang. mixture of multivariate normal distributions). To však může zavést závažné zkreslení.

Mean-shift je založen na neparametrické analýze příznaků, která tyto zkreslené příznaky vylučuje. Pracuje se s funkcí hustoty pravděpodobnosti ležící pod  $n$ -dimenzionálním prostorem příznaků. Z této funkce se odhadnou mean shift vlastnosti, ze kterých se určí gradient hustoty v bodě. Na základě toho se z mean-shift stane detektor *módusů*, což jsou maxima funkce hustoty. Tak se děje v okně, které se v každé iteraci posune blíže k pozici, kde je nejhustší zastoupení módusů, viz obrázek 2.3. Tím se získají středy segmentů, ze kterých se následně určí celý segment. Oproti K-means není tato metoda omezena počtem segmentů. Výhodou tohoto přístupu je, že s pomocí získaných dat z obrazu lze získat detektor hran v obraze. Ten dle [3], by měl být kvalitnější než metody založené pouze na gradientu v obraze.



Obrázek 2.3: Ukázka práce Mean-Shift, hledání hustých módusů

## Kapitola 3

# Výpočet disparity

Získání přesné mapy disparit je náročné až nemožné ze tří důvodů:

1. Až polovina bodů ve scéně nemá s čím korespondovat. Korespondující bod v druhém snímku neexistuje z důvodu, že je zakryt jinou částí scény, nebo byl sloučen s jiným bodem kvůli změně úhlu pohledu.
2. V obraze může být hodně *šumu*, což jsou náhodné rušivé hodnoty. Ty znemožňují správné určení korespondencí.
3. Ideálně by se měl objekt scény v konstantní vzdálenosti od objektivu zobrazit jako stejně jasná, stejně barevná oblast. To se však v praxi neděje a obraz je pořízen s rozdílnou intenzitou osvětlení a barvou pro tyto oblasti. To může vést k dalším chybám.

Pro pochopení další části je třeba definovat pojmy *cena*, *energie*, *značkování* a *hladkost*

**Cena:** Operací mezi bodem  $p_p$  a  $p_l$ , např. absolutní rozdíl hodnot bodů, se určí hodnota, které se říká cena. Cena odráží svou hodnotou odlišnost bodu  $p_p$  a  $p_l$ . Ideálně by korespondující body scény měly mít cenu nulovou.

**Značkování:** Značkování je funkce definující, jaká disparita kterému bodu byla přidělena. Většinou se začíná se značkováním, které vybírá pro každý bod disparitní mapy disparitu s minimální cenou.

**Hladkost:** Z důvodu, které jsou na začátku této kapitoly, se může stát, že bod v obraze je špatně označován. Tyto chyby se řeší tak, že kromě minimální ceny se hodnotí i hladkost obrazu. Hladkost je tedy ocenění, čím menší tím lépe, které hodnotí aktuální stav značkování a penalizuje velké změny disparity.

**Energie:** Hladkost a minimální cena dohromady tvoří energii značkování. Většina optimalizačních metod se snaží tuto energii co nejvíce zmenšit. Minimalizace energie však nesmí být na úkor ztráty struktury scény. Proto obvykle dochází k tomu, že se vybere vyšší cena značkování s nižší hladkostí pro snížení celkové energie.

### 3.1 Globální metody

Jsou to takové metody, které provádí skoro všechny své operace při výpočtu disparity. Hned je vidět výsledek. Spousta těchto metod se zaměřuje na techniku energetické minimalizace, jako jsou např. řezy grafu, belief propagation, dynamické programování, scanline optimalizace nebo simulované žíhání [7], [9]. Oproti lokálním metodám pro výpočet disparity se uvažují i stavy ostatních bodů, nebo segmentů.

#### Belief propagation

Jsou algoritmy pro vyhodnocení MRF, Markovské náhodné pole (ang. Markov Random Field). Základem pro pochopení je, že obraz je chápán jako graf, kde jednotlivé uzly jsou body, či segmenty a hrany, spojující sousedy mezi sebou. Zde každý uzel posílá zprávy přes všechny své hrany. Každá taková zpráva reprezentuje přesvědčení odesílatele o příslušnosti příjemce k disparitě. Výsledkem jsou hodnoty věrohodnosti disparit od všech sousedů v grafu, ty udávají přesvědčení okolí o příslušnosti k disparitě.

Jde o iterativní výpočet, ale v každém kroku je nutné provést výpočet  $N^2$  možných řešení pro každý bod, kde množina  $N$  je počet možných značek, které mohou být uzlu přiděleny.

#### Řezy grafu

Stejně jako BP, Belief Propagation, jsou to metody pro minimalizaci energie grafu. Pracuje se zde s tzv. *prohazovacím algoritmem*, jde o výměnu mezi dvěma možnými stavy značek a *algoritmy maximálního protékání*, maximalizuje se tok mezi dvěma vrcholy, kterým se říká původní vrchol (ang. source) a vnořený vrchol (ang. sink). Řezy grafů dovedou rychle aproximovat, rychleji než BP, do přijatelného řešení [8].

#### Dynamické programování

Zatímco u 2D optimalizace energetické funkce globální metody se může ukázat jako NP-těžké (nedeterministicky polynomiální čas) pro běžnou třídu plynulosti. Dynamické programování umí nalézt nezávisle *scanline*, řádky v obraze, které si odpovídají, v polynomiálním čase. Poprvé byla metoda použita pro stereo vidění v metodě, založené na detekci hran [9].

#### Scanline optimalizace

Metody využívající model ceny lineární hladkosti. Hlavní částí metody je počítání vhodné minimální konvoluce, která je obvykle implementována jako dvouprůchodový algoritmus, užívající *destruktivní aktualizace pole*. Změny se dělají namísto rekurzivním postupem upraveným pro paralelní zpracování. Metody jsou uzpůsobeny, aby je bylo možno provádět na GPU, procesor umístěný na grafické kartě zajišťující vykreslování. Experimenty s těmito metodami říkají, že jsou vhodné pro výpočty na snímcích s vysokým rozlišením a snímcích s nesourodou disparitou [13].

#### Simulované žíhání

Jsou metody, které zabraňují uváznutí při řešení optimalizace funkce v lokálním minimu. Tomu se vyhýbají počátečními velkými změnami. Klasické algoritmy přijímají pouze řešení,

které je lepší. Optimalizace simulované žíhání přijímá i horší řešení. Tím se algoritmus dostane z lokálního minima a může nalézt globální minimum.

## 3.2 Lokální metody

Oproti globálním metodám kladou důraz jen na hodnoty barvy a intenzity bodu snímku pro výpočet disparity. Tak se děje v osmi-okolí pro každý bod. Stává se, že lokální metody po ukončení svoji práce provedou agregace a výpočet začne od začátku. Děje se tak proto, aby se zvýšila kvalita. Způsobů výpočtů je několik, mají však společný rys, že vytváří cenu výpočtu, ze které se až následně vypočítá disparitní mapa. Jednou z možných metod získávání disparitní mapy z cen disparit, je winner-take-all metoda. Lokální metody lze rozdělit do tří skupin podle toho, co je jejich základem pro výpočet. Existují metody založené na seřazení, gradientu a intenzitě [7].

### Seřazení

Jde o velice robustní metody, mající dobré výsledky při měnícím se osvětlení. Způsob jak tyto metody pracují je, že se hledají kombinace bodů, střed okna vůči jeho osmi-okolí, a to na stejných pozicích jak v pravém tak v levém obraze, které se rovnají. Při měření výkonu ostatních dvou druhů se využívá tato jako referenční. Metody tohoto typu jsou založeny na následující definici.

$$C_{census}(i, d) = \sum_{(x,y) \in N+(P_i)} p(x, y, d) \quad (3.1)$$

$$p(x, y, d) = \begin{cases} 0 & \text{if } L_{x,y} > L_i \quad \text{a} \quad P_{x-d,y} > P_{i-d} \\ 0 & \text{if } L_{x,y} < L_i \quad \text{a} \quad P_{x-d,y} < P_{i-d} \\ 1 & \text{ostatní} \end{cases} \quad (3.2)$$

Zde  $N$  je osmi-okolí bodu v  $L$  (levém) obraze,  $p_1$  je bod ve středu osmi-okolí  $N$ ,  $L_{x,y}$ , resp.  $P_{x-d,y}$ , je bod z  $N$  v obraze  $L$ , resp.  $P$ . A  $L_i$  a  $P_i$  jsou body ze středu  $N$  v obraze  $L, P$ .

### Gradient

Tyto metody využívají pro měření podobností prostorovou vzdálenost mezi koncovými body vektoru gradientu. Vyjadřují se zde rozdíly ze středu počítaného okna. Tím, že se počítá s hodnotou ze středu, se udržuje vliv sousedů uvnitř okna.

$$C_{GRAD}(i, d) = |\nabla L_i - \nabla P_{i-d}|_1 \quad (3.3)$$

Zde  $\nabla$  představuje odhad použitím středních rozdílů a  $|*|_1$  je normalizační funkce vůči levému obrazu.

### Intenzita

Nejjednodušším představitelem je počítání *absolutních rozdílů*, budou-li se porovnávat výsledky s gradientními nebo seřadovacími metodami. Pro spravedlivé porovnání je vhodné použít *sumu absolutních diferencí* nad oknem větším, jak se uvažuje při počítání gradientu nebo seřazení.

$$C_{AD}(i, d) = |L_i - R_{i-d}| \quad (3.4)$$

$$C_{SAD} = \frac{1}{|N|} \sum_{(x,y) \in N+(p_i)} |L_{x,y} - R_{x-d,y}| \quad (3.5)$$

$C_{AD}$  jde o nejjednodušší a nejlevnější, ve smyslu nároků na výpočet, operaci. Ta měří absolutní rozdíl v levém a pravém obraze, který je patřičně posunut. V  $C_{SAD}$  stejně jako v  $C_{AD}$  se měří absolutní rozdíl, zde se tak děje v okně  $3 \times 3$ .

## Kapitola 4

# Analýza metod pro výpočet disparity

Metody popsané v předchozí kapitole jsou základní. I když jsou schopny vyjádřit disparitní mapu, jejich kvalita je nízká [11]. V praxi se používají kombinace těchto metod, ale obohacují se o metody, které byly vytvořeny pro jiné obory počítačového vidění. Vždy se však jedná o lepší výsledky, než jakých je možno dosáhnout použitím jenom těchto základních metod.

Určitý čas se nevedly záznamy a porovnání mezi jednotlivými metodami. Tento stav však nebýval překážkou, avšak kolem roku 2000 začalo vznikat množství prací, zabývajících se problematikou stereovidění. Pak již bylo těžké se v těchto metodách vyznat a srovnávat je.

Poměrně brzy na to reagovala Middleburská universita v americkém Vermontu. Zde vznikla práce [9], která položila základy pro srovnávání metod a vytvořila standardizovaný postup pro jejich porovnávání. Jejich výsledky se od té doby zaznamenávají a jsou přístupné na webových stránkách univerzity věnovaných stereovidění [4].

Práce a web se staly výchozími zdroji při výběru metody pro získávání disparitní mapy. V době výběru metody, kterou tato práce bude popisovat, byla metoda AdaptingBP[3] na druhé pozici. První bývala ADCensus[12], jde však stále o poměrně novou metodu a materiálů pro její důkladnější pochopení nebylo tolik jako pro druhou nejlepší metodu AdaptingBP. Dnes v době dokončování této práce již obě metody byly překonány metodou AdaptGCP. Ta byla do žebříčku přidána poměrně nedávno.

Jako hlavní zdroj této práce pro výpočet disparitní mapy se stal článek o metodě AdaptingBP, celým názvem: *Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure*. (česky Metody výpočtu disparity založené na segmentaci, adaptabilním měření odlišnosti a šíření přesvědčení)

## Kapitola 5

# Metoda založená na segmentaci a optimalizaci

Tyto metody se vyznačují dobrým *výkonem*, poměr mezi rychlostí a kvalitou. Jako základ mají předpoklad, že každý snímek lze převést na roviny, kde každá rovina odpovídá aspoň jednomu barevnému segmentu v referenčním snímku. Metody obsahují 4 kroky:

1. Segmentace obrazu na úseky se stejnou intenzitou a barevností.
2. Hledání korespondujících bodů. Používají se lokální metody pro nalezení disparity mezi body, které mají velkou šanci, že reprezentují ten samý bod scény.
3. Ze získaných segmentů a korespondencí se vytváří roviny disparity. Kroku se říká přiléhání rovin (ang. fitting). Získáme tím kolekci rovin stejného počtu jako je segmentů.
4. Optimalizace získaných rovin se aproximuje hladovými algoritmy či řezem grafů.

Metoda popsána v této práci obsahuje též tyto čtyři kroky, jsou zde však tři úpravy:

1. Hledání korespondencí se během výpočtů přizpůsobuje pro navýšení počtu věrohodných korespondencí bodů.
2. Nový přístup pro získání disparitních rovin
3. Optimalizace značkování se řeší metodou Loopy Belief Propagation

### 5.1 Barevná segmentace

Prvním krokem je rozdělení referenčního snímku na stejnobarevné oblasti nebo oblasti se stejným odstínem šedi. Předpokladem je, že v takovéto oblasti se disparita výrazně nemění, nebo se mění plynule. Je zde pravidelnost ve změnách. Tento předpoklad je založen na odhadu, že k velkým změnám může dojít pouze na hranicích segmentů. Pro tuto část byla použita práce Comariciu a Meer [3]. Podrobnější popis je v sekci zpracování obrazu 2.3



## 5.2 Hledání korespondencí

V popisované metodě se určuje disparita  $d$  pro každý bod. Z těch se pro každý segment odhadne rovina. K tomu účelu je třeba určit tři koeficienty takové roviny  $c_1, c_2, c_3$ . Tyto koeficienty pak budou vyjadřovat disparitu v segmentu podle rovnice roviny:

$$d = c_1x + c_2y + c_3. \quad (5.1)$$

Zde jsou  $x$  a  $y$  souřadnice polohy bodu.

Tímto postupem se získá velké množství rovin. V další části bude ukázáno, jak množství rovin zmenšit na dostačující množství na zachování struktury scény bez její ztráty.

Prvním krokem je výpočet ceny disparity pro všechny body v obraze. V popisované metodě se používá kombinace metody sumy absolutních diferencí  $C_{SAD}$  a metoda měřící gradient  $C_{GRAD}$ . Jednotlivá měření jsou definovány jako:

$$C_{SAD}(x, y, d) = \sum_{(i,j) \in N(x,y)} |I_1(i, j) - I_2(i + d, j)| \quad (5.2)$$

$$C_{GRAD}(x, y, d) = \sum_{(i,j) \in N_x(x,y)} |\nabla_x I_1(i, j) - \nabla_x I_2(i + d, j)| + \sum_{(i,j) \in N_y(x,y)} |\nabla_y I_1(i, j) - \nabla_y I_2(i + d, j)| \quad (5.3)$$

Zde  $N(x, y)$  je okno  $3 \times 3$  se středem v bodě  $(x, y)$ ,  $N_x(x, y)$  je  $N(x, y)$  bez nejpravějšího sloupce. Obdobně  $N_y(x, y)$  je  $N(x, y)$  bez nejspodnějšího řádku.  $\nabla_x$  je filtr pro detekci horizontálních hran,  $\nabla_y$  detekce vertikálních hran.

Výsledná míra ceny se spočítá jako:

$$C(x, y, d) = (1 - \omega) \cdot C_{SAD}(x, y, d) + \omega \cdot C_{GRAD}(x, y, d) \quad (5.4)$$

Hodnota  $\omega$  se určí maximalizováním počtu shodných disparit mezi výpočty zprava doleva a zleva doprava. Tato operace se provádí s metodou winner-take-all, která spočívá v tom, že se vybere pro každý bod ta disparita, jenž má nejmenší cenu.

## 5.3 Odhad disparitních rovin

Přestože při výpočtu ceny se provádí kroky pro minimalizaci chyb, chyby se nepodaří zcela odstranit. Proto dalším krokem je odhad roviny disparity segmentu, při tomto výpočtu bude snaha se těmito chybám vyhnout a nezapočítávat je do výsledku. K tomu účelu se použije Gaussův filtr, který určí váhy tak, aby se věrohodné hodnoty zachovaly a lokální extrémny odstranily.

Počítají se koeficienty  $c_1, c_2, c_3$  do rovnice 5.1. Postup při výpočtu koeficientu  $c_2$ , čili vertikální koeficient, je uveden v Algoritmu 1 níže.

---

**Algoritmus 1:** Výpočet  $c_2$ 

---

```
inicializace vektoru difference  $\vec{V}$ 
for Všechny segmenty do
    Pro každý jeden sloupec segmentu se vypočítají difference v disparitě
    Získaný vektor se připojí k vektoru  $\vec{V}$ 
end for
seřadíme vektor  $\vec{V}$ 
Na seřazený vektor je aplikován Gaussův filtr
Tím je získán koeficient  $c_2$ 
```

---

Parametr  $c_1$  je počítán obdobně, ale pro jednotlivé řádky. Parametr  $c_3$  se vypočte z rovnice 5.1, kde za  $d$  se dosadí odhad disparity ze středu segmentu. To je použitím již získaných odhadů  $c_1$  a  $c_2$  v upravené rovnici roviny. Ta nyní vyjadřuje  $c_3$  a za  $d$  se dosadí všechny disparity. Na získaný vektor se použije Gaussův filtr.

Pro zvýšení přesnosti se provádí krok, kdy se seskupují segmenty. Seskupují se ty segmenty, které mají stejnou rovinu. Před seskupováním se zkoumá, zdali jiná rovina pro konkrétní segment, by nebyla vhodnější, tj. měla menší cenu. Operaci se říká minimalizování ceny a vybírá se ta rovina, značka, které má nejmenší hodnotu pro každý segment zvlášť. Tato cena se počítá jako

$$C_{SEG}(S, P) = \sum_{(x,y) \in S} C(x, y, d). \quad (5.5)$$

Kde  $P$  je disparitní rovina a  $S$  je segment.

Z této ceny se určí minimální cena pro segment. Dále se segmenty, které mají společné značkování, seskupí do jednoho segmentu. Pro nově vzniklé segmenty se znovu spočítá odhad rovin. Ten bude použit v optimalizaci jako parametr  $E_{data}$ .

## 5.4 Optimalizace

Vzniklé minimální značkování stále přesně nemusí odpovídat realitě. Je tomu tak u segmentů, které mají velký počet chyb vůči svojí velikosti a nebyla jim proto nalezena správná rovina.

Proto se v závěrečném kroku provádí minimalizace energie celého značkování  $f$ . Ta počítá kromě ceny také s hladkostí mezi segmenty. Energie se počítá jako

$$E(f) = E_{data}(f) + E_{smooth}(f). \quad (5.6)$$

Zde

$$E_{data}(f) = \sum_{s \in R} C_{SEG}(s, f(s)) \quad (5.7)$$

$$E_{smooth}(f) = \sum_{\forall (s_i, s_j) \in S_N | f(s_i) \neq f(s_j)} \lambda(s_i, s_j, f(s_i), f(s_j)) \quad (5.8)$$

Množina  $R$  představuje všechny segmenty a  $s$  konkrétní jeden segment. Funkce  $f$  je značkovácí funkcí. Množina  $S_N$  reprezentuje všechny sousedy a  $\lambda(s_i, s_j, f(s_i), f(s_j))$  je penalizace mezi sousedními segmenty na základě délky hranice a střední hodnoty barvy.

Proměna  $E_{smooth}$  je funkce, ze které se vytvoří pro lepší manipulaci matice. Funkce  $\lambda(s_i, s_j, f(s_i), f(s_j))$  je definována jako suma součinnů [1]:

$$\lambda(s_i, s_j, f(s_i), f(s_j)) = \sum_{s_i, s_j \in S \wedge (s_i, s_j) \in NB} \begin{cases} \lambda_{disc} \cdot borderlen(s_i, s_j) \cdot coloursimilarity(s_i, s_j) & : f(s_i) \neq f(s_j) \\ 0 & : \text{otherwise} \end{cases} \quad (5.9)$$

$s_i$  a  $s_j$  jsou segmenty,  $NB$  je množina všech segmentů, které spolu sousedí.  $\lambda_{disc}$  je kladná konstanta penalizace segmentů, které narušují plynulost. Její hodnota není nikde přesně definována, a v této implementaci byla vždy nastavována na  $1/10$  průměrného přírůstku ceny v řádku matice cen ( $E_{data}$ ).

Funkce *coloursimilarity* porovnává barevnou podobnost dvou segmentů. Funkce je definována jako [1]:

$$coloursimilarity(s_1, s_2) = \left(1 - \frac{\min(|meancolor(s_1) - meancolour(s_2)|, 255)}{255}\right) \cdot 0.5 + 0.5 \quad (5.10)$$

Obor hodnot této funkce je  $< 0.5, 1 >$ . Pro barevně stejné segmenty funkce vrací hodnotu 1, pro segmenty které mají rozdíl v barvě větší nebo roven 255, funkce vrací hodnotu  $1/2$ .

Optimalizace značkování se provádí minimalizací energie za pomoci Loopy Belief Propagation, kde segmenty posílají zprávy o svém přesvědčení o příslušnosti segmentu k rovině.

## 5.5 Loopy belief propagation

O způsobu jakým metoda pracuje, bylo řečeno, že jde o posílání zpráv mezi segmenty o přesvědčení odesílatele o příslušnosti příjemce ke značce. Odesílající segment posílá zprávu  $m_{s_i \rightarrow s_j}^t$  sousednímu segmentu. Ta má podobu:

$$m_{s_i \rightarrow s_j}^t = \min_{f(s_i)} \left( \lambda(s_i, s_j, f(s_i), f(s_j)) + C_{SEG}(s_i, f(s_i)) + \sum_{s_k \in N(s_i) \setminus s_j} m_{s_k \rightarrow s_i}^{t-1}(f(s_i)) \right) \quad (5.11)$$

Funkce  $\lambda$  je funkcí  $E_{smooth}$ , jsou zde však dva parametry navíc. Ty jsou tu proto, aby se podmínka výrazu 5.9 naplnila a segmenty se stejnou funkcí budou penalizovány nulou. Další funkcí je  $C_{SEG}$ . Jde o matici cen nad seskupenými segmenty z kapitoly 5.3. Suma na konci říká, že na rozhodování o tom, jakého přesvědčení jsou odesílatele, má vliv i přesvědčení jejích sousedů o příslušnosti v minulé iteraci, avšak mimo přesvědčení segmentu, kterému bude zpráva odeslána. Celý tento výraz je pak uzavřen ve funkci  $\min_{f(s_i)}$ . Která stnovuj, že se má vybrat minimální hodnotu pro všechny značky, v nich se může odesílatel vyskytnout. To jsou všechny značky. Ukazuje se, že zpráva nebude pouze jedna hodnota, ale vektor tolika hodnot, kolik je různých značek. To proto, že odesílatel sděluje svoje přesvědčení o příslušnosti ke každé značce zvlášť.

Příklad komunikace mezi Odesílatel=O Příjemce=P přičemž existují 3 různé značky:

O: Za předpokladu, že já jsem značka 1, pak si myslím, že P je značka  $[1, 2, 3]$  s hodnotou přesvědčení  $[30_{1a}, 50_{2a}, 10_{3a}]$ .

O: Za předpokladu, že já jsem značka 2, pak si myslím, že P je značka  $[1, 2, 3]$  s hodnotou přesvědčení  $[40_{1b}, 8_{2b}, 12_{3b}]$ .

O: Za předpokladu, že já jsem značka 3, pak si myslím, že P je značka  $[1, 2, 3]$  s hodnotou přesvědčení  $[100_{1c}, 9_{2c}, 21_{3c}]$ .

Z výsledné komunikace se vybere z každé zprávy minimální hodnota. To znamená, že zpráva bude vypadat:  $\text{Msg} = [10_{3a}, 8_{2b}, 9_{2c}]$ . Touto zprávou odesílatel sdělil, že jeho největší předpoklad je, že segment by měl mít značku 2.

## Kapitola 6

# Implementace

Zde se budou podrobněji popisovat kroky z kapitoly 5 pro pochopení prováděných operací. Pro samotnou implementaci byl vybrán program Matlab, který je dobře uzpůsobený pro práci s obrazem a maticemi obecně. Proto se často v popisu metod zmiňuje užitečnost zaznamenávat a pracovat nad celými maticemi namísto iterativního přístupu bod po bodu.

### 6.1 Barevná segmentace

Metody segmentace nejsou zcela jednoduché a není to náplní této práce. Proto bylo použito již hotového projektu od Shai Bagon [10], který naimplementoval pro Matlab projekt EDISON z Rutgerské univerzity [2]. Z tohoto programu se využije výstupu labels, značky. Ty mají pro tento účel přesný formát. Šedotónový obraz, kde každý bod v segmentu má stejnou hodnotu a hodnota každého segmentu je unikátní.

### 6.2 Hledání korespondencí

První krok je jednoduchý, spočívá v naimplementování měření  $C_{SAD}$  a  $C_{GRAD}$  podle rovnic 5.2 a 5.4. Dalším krokem je učít nejlepší hodnotu  $\omega$ . O té se říká, že by měla maximalizovat počet shodných disparit při počítání z levého obrazu k pravému, tedy se posune pravý obraz doleva a z pravého obrazu k levému, zde se posune levý obraz doprava<sup>1</sup>. Co je neznámé, je disparitní mapa. Jde o mezivýsledek získaný na základě metody winner-take-all.

#### Winner-take-all

Jak již bylo naznačeno, jde o hledání minimální ceny. Z výpočtu  $C_{SAD}$  a  $C_{GRAD}$  získáváme matici, která sděluje jaká je cena pro každý bod z levého obrazu pro konkrétní disparitu  $d$ . Bude-li se uvažovat 31 disparitních úrovních (disparita 0-30) vznikne 31 cenových obrazů. Cenových obrazů je tedy stejné množství, jako je uvažovaných disparit. Vznikla tím třídimenzionální matice se souřadným systémem  $(x, y, d)$ .  $x$  a  $y$  jsou pozice bodu v obraze, souřadnice  $d$  udává disparitu.

---

<sup>1</sup>Protože při počítání z pravého obrazu k levému je obrácený směr, hodnoty vyjdou záporné. Musí se proto hodnoty převrátit.

Nyní jsou všechny potřebné informace známy a lze tak zjistit minimální ceny pro každý bod a určení tzv. „hrubé disparity“.

---

**Algoritmus 2:** Výpočet hrubé disparity

---

```

 $dis(\cdot) = 0$ 
for  $\forall p(x, y) \in I_1$  do
     $minZ = \infty$ 
    for  $\forall z$  do
        if  $M(x, y, z) < minZ$  then
             $minZ = z$ 
        end if
    end for
     $dis(x, y) = minZ$ 
end for

```

---

Nyní se bude hledat  $\omega$ , která maximalizuje počet shodných disparit. Jde opět o jednoduchý algoritmus, se zvoleným krokem budeme navyšovat hodnotu  $\omega$  a zaznamenávat počet shodných hodnot z „hrubých“ map disparit v obou směrech. S nejlepší hodnotou  $\omega$  se spočítá cenová třídímní matice, podle 5.4, a „hrubá“ disparita. „Hrubá“ disparitní mapa bude využita při operaci přiléhání rovin, cenová matice při optimalizaci.

### 6.3 Odhad disparitních rovin

Zde bude popis výpočtů koeficientů  $c_1, c_2, c_3$  pro rovnici 5.1 a ceny segmentů pro jednotlivé značky disparitních rovin. Samotný algoritmus pro získání koeficientů byl již dobře popsán v kapitole 5. Při implementaci bylo pozorováno, že optimální nastavení Gaussova filtru je rovno  $\sigma$  rovna  $1/8$  délky vstupního vektoru.

Suma ceny v segmentu se počítá dle vzorce 5.5. Bude-li se záznam o cenách zaznamenávat do matice 5.5 kde osa  $x$  bude reprezentovat značku a osa  $y$  segment. Celý proces získávání takovéto matice je zdoluhavý, může trvat až stovky sekund. Při podrobnějším prohledání matice se ukazují duplicitní řádky. Z toho vyplývá, že koeficienty takových rovin musí být stejné nebo si velice blízké. Zkoumáním koeficientů těchto rovin se zjistilo, že rozdíl v hodnotách menších jak  $10^{-2}$  neovlivňuje výsledek. Proto bylo učiněno rozhodnutí zaokrouhlit všechny hodnoty na jednu tisícinu a odstranit duplicity. Tímto krokem se ve většině případů zredukovalo množství značek o více jak 70 %. Zároveň se tím zredukoval čas nutný pro výpočet, ve většině případů na dobu do 40 sekund. Další redukce může nastat po vytvoření matice cen, zde se provádí minimalizace, kdy se hledá pro každý segment ta rovina, která pro tenot segment bude mít nejmenší cenu. Může se tak stát, že určité rovině budou odebrány všechny segmenty a stane se tím již nepotřebnou. Naopak segmenty, kterým je přiřazena stejná rovina, budou seskupeny.

Celý proces odhadování rovin se provede ještě jednou nad seskupenými segmenty. Tím se získá nový odhad roviny pro seskupené segmenty a bude přepočítána matice  $C_{SEG}$  pro nově vzniklé roviny nad původními segmenty. Této matice se využije při optimalizaci.

## 6.4 Optimalizace

Dle kapitoly 5 se optimalizace provádí minimalizací energie značkování 5.6. Energie se tedy počítá jako součet  $E_{data}$  a  $E_{smooth}$ . Dle definice se vypočítá proměnná  $E_{data}$  stejně jako cena značkování  $C_{SEG}$  (5.5). Za tuto proměnnou tedy se bude dosazovat sumu cen ze segmentu při konkrétním značkování. K tomuto účelu se využívá již vypočítané matice cen z výpočtu v kapitole 5.3. Jde o tabulku s rovinami vypočítanými v druhém kole odhadu nad seskupenými segmenty. Z této tabulky budou vybrány ty hodnoty cen podle aktuálního značkování.

Před samotnou optimalizací je však třeba dodefinovat funkce *borderlen*. Jak název napovídá, měří délku hranice. Měření je v bodech mezi segmenty. Implementací této funkce existuje několik, v této implementaci byla použita metoda, kdy se vytvoří zvětšená maska segmentu o jeden bod ve 4 směrech. Od této masky se odečte původní maska segmentu. Vznikne maska vnějších hranic segmentu. Tato maska se použije na výstup labels z barevné segmentace, tím se získala sousednost segmentů. Pak již stačí jen spočítat délku hranice u každé dvojice sousedů. Údaje o délce společné hranice je vhodné zaznamenávat do tabulky.

Při výpočtu funkce *coloursimilarity* se počítá s nedefinovanou funkcí *meancolour*. Jde o výpočet průměrné barvy podle vzorce

$$Meancolour(S) = \frac{\sum_{p \in S} p_R + p_G + p_B}{size(S)}, \quad (6.1)$$

kde  $p_R, p_G$  a  $p_B$  jsou hodnoty barevné složky červená, zelená a modrá bodu  $p$  ze segmentu  $S$ . Funkce  $size(S)$  vrací počet bodů v segmentu.

Ve výrazu o výpočtu hladkosti se udává podmínka  $f(s_i) \neq f(s_j)$ , ta sděluje, že hladkost se počítá pouze mezi sousedními segmenty, které mají přiřazeny různé značky disparitních rovin. V ostatních případech je hodnota 0. Při optimalizaci se bude měnit značkování segmentů pro snižování energie. Určit tedy, které segmenty mají stejnou značku, je v tuto chvíli nemožné. Proto se bude tato podmínka prozatím ignorovat a bude se postupovat jako by všechny segmenty měly různé značky. Podmínka bude naplněna v průběhu optimalizace. Při každém novém značkování se vymaskují ty hodnoty, jenž splňují podmínku  $f(s_i) \neq f(s_j)$ . Sumou těchto hodnot získáme hladkost aktuálního značkování.

## 6.5 Loopy belief propagation

V samotné metodě nebyly při implementaci učiněny žádné změny a byla implementována dle kapitoly 5.5.

## 6.6 Výpočet hloubky

Po optimalizaci jsme získali optimalizované značkování. Posledním krokem je přepočtení disparity na hloubku. V kapitole 2 je uvedena rovnice 2.1 přepočtu disparity na hloubku, kde vystupují parametry ohnisková vzdálenost  $f$ , disparita  $d$  a vzdálenost mezi objektivy při pořizování obrazů  $|C - C'|$ , tzv. *báze*. Báze je zaznamenána při pořizování obrazů. Disparitní mapa byla vytvořena v předchozím kroku. Ohnisková vzdálenost je známá, ale v mm. Disparita se však uvádí v bodech. Proto je třeba učinit převod ohniskové vzdálenosti z mm na body. Pro převod se využije rovnice

$$f_p = \frac{f}{width_{CCD}} \cdot width_{image}, \quad (6.2)$$

kde  $f_p$  je ohnisková vzdálenost v bodech,  $f$  ohnisková vzdálenost v mm,  $width_{CCD}$  je šířka CCD čipu v mm a  $width_{image}$  je šířka obrazu. Tím se získalo ohnisko vyjádřená v bodech. Aby bylo možno dále pokračovat, bylo nutné zjistit rozměry čipu CCD a fotografie. Fotografie byly pořizovány fotoaparátem Nikon D50, který má velikost čipu  $23.7 \times 15.6mm$  a v nejvyšším rozlišení vytváří snímky  $2000 \times 3000$  bodů. Bude-li během výpočtu změněna velikost fotografie, musí se přepočítat i ohnisková vzdálenost. Jde o přímou úměru, stačí násobit správným koeficientem.



## Kapitola 7

# Testování a diskuze

V této kapitole se bude implementované řešení testovat na kvalitu tvorby disparitních map a hloubkových map. Pro testování disparitních map byla použita kolekce obrazů z webu Middleburské university věnované stereovidění [4]. Zde je umístěno šest kolekcí obrazů. V každé kolekci je kromě devíti snímků také obraz *ground true disparity*, ten ukazuje přesnou disparitu v obraze. Vzniklé disparitní mapy budou porovnávány s „ground truth disparity“. K porovnání bude použit postup, kdy se ve snímku zobrazí červeně místa, kde disparita byla určena špatně o více jak jeden bod.

Hloubková mapa bude testována s údaji naměřenými při pořizování obrazu.

Schéma obrazů bude vždy stejné.

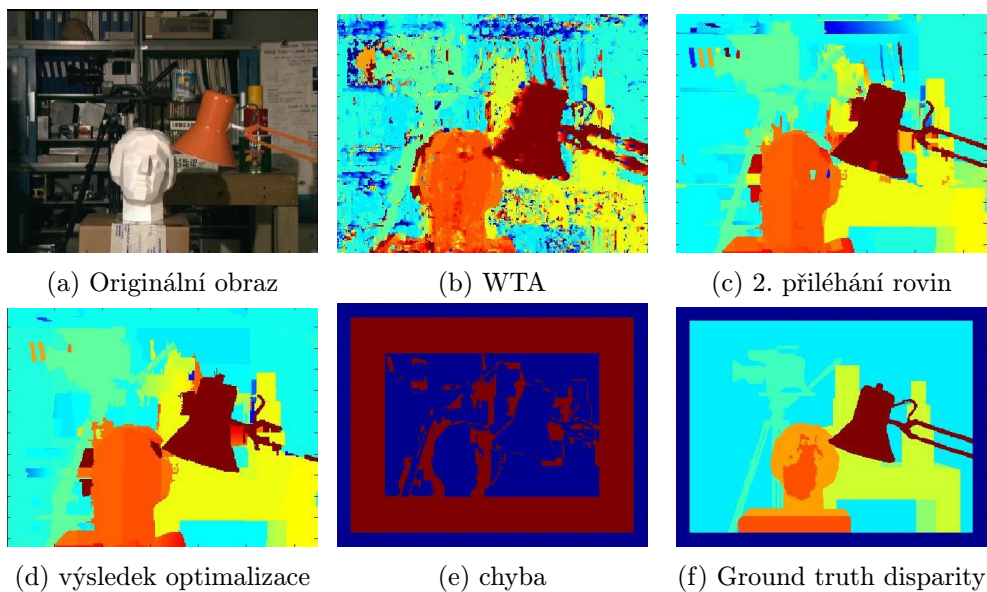
### Jednotlivé obrazy:

- a) Originální obrázek.
- b) WTA: winner take all, prvotní změřená „hrubá“ disparita.
- c) 2. přiléhání rovin: jde o výsledek kdy se počítají roviny pro každý segment.
- d) výsledek optimalizace: po deseti interakcích optimalizace Loopy BP.
- e) chyba (pouze u disparitní mapy): rozdíl mezi ground truth disparity f) a optimalizované disparity d)
- e) hloubková mapa (pouze u hloubkové mapy): mapa reprezentující hloubku v obraze.
- f) Ground truth disparity (pouze u disparitní mapy): Jde o disparitní mapu, která přesně odpovídá scéně.

V této kapitole budou ukázány testy od nejzajímavějších obrazů, zbylé testy ve stejném rozložení budou v příloze B.

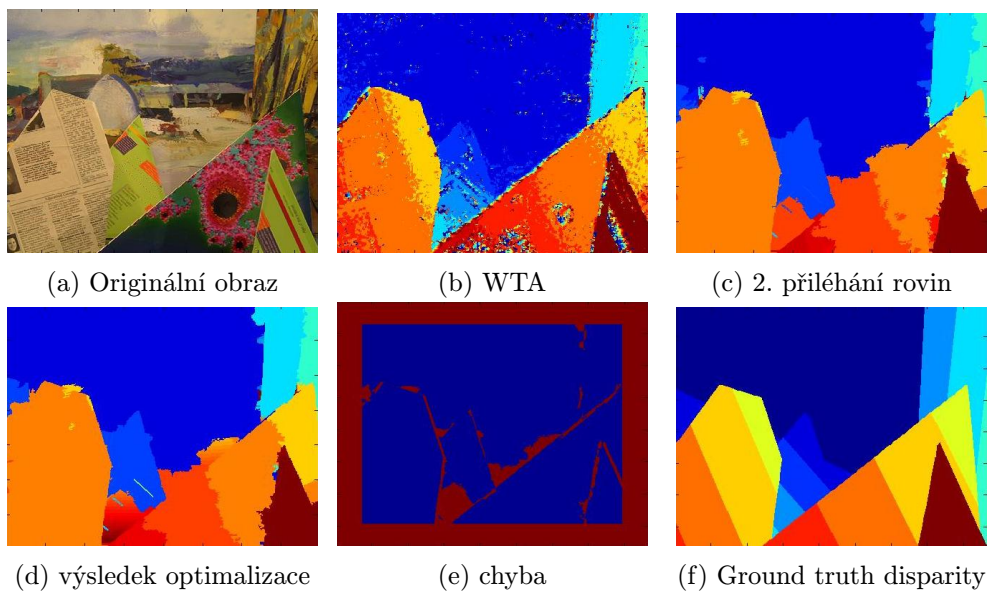
### Disparitní mapa

Disparitní mapy budou testovány na obrazech z webu [4]. Kromě kolekce Tsukuba jde vždy o obraz 2. a 6. U Tsukuba je disparitní mapa mezi 2. a 5. obrazem. Vzniklá disparitní mapa bude porovnána „ground truth disparity“.



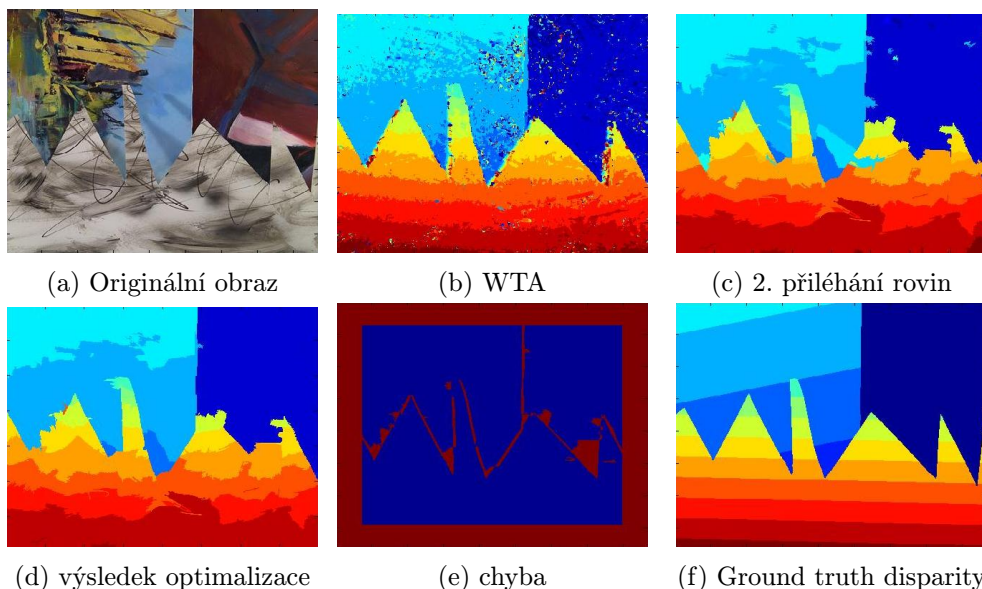
Obrázek 7.1: Tsukuba

Na kolekci obrázků 7.1 je vyobrazení často používané scény busty, lampy a kamery. Implementována metoda dle obrazu nazvaným chyba měla mnoho chybných klasifikací disparity. K tomu se v obraze vyskytuje obdélník, který je způsoben chybným ořezáním obrazu od neužitečných dat. Implementovaná metoda sice jeví velké chyby v určování hranic objektů, avšak určila správně disparity významných objektů scény. To jsou lampa, stůl, kamera a busta.



Obrázek 7.2: Barn1

V druhé kolekci obrazů 7.2, je vybrána scéna, ve které byl jako v druhé neměřen nejmenší počet chyb, tj. špatné určení disparity o víc jak 1 disparitní stupeň.



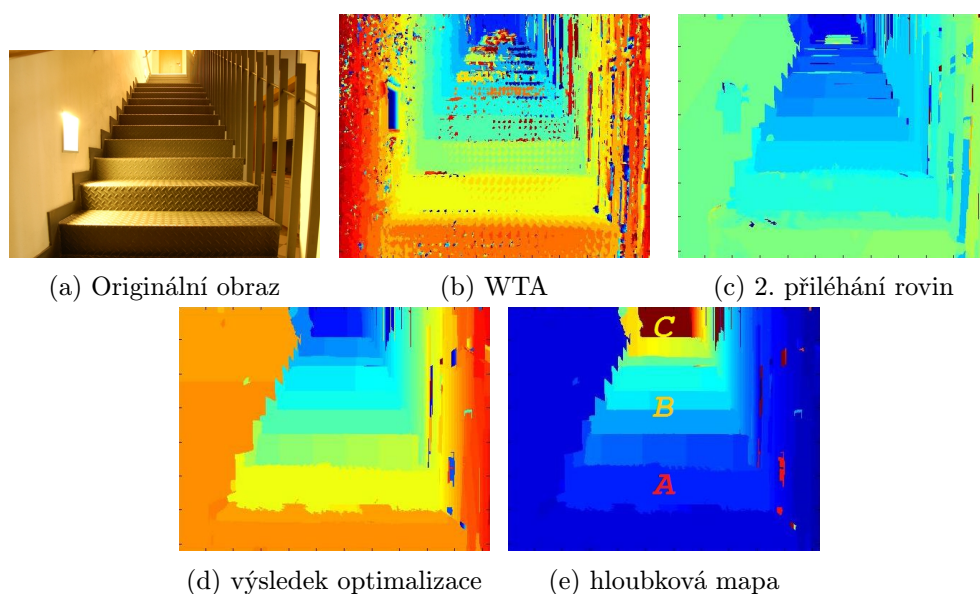
Obrázek 7.3: Saw

Poslední kolekce nese název Saw. Zajímavý je na této kolekci fakt, že i když snímky pohledově vykazovaly horší nebo srovnatelný výstup s ostatními kolekcemi, výsledná chybová mapa je nejmenší. Chyby byly totiž v toleranci jednoho disparitního stupně a větší část bodů tedy byla klasifikována jako hodnota odpovídající referenční. V příloze A je umístěno dalších 6 testů tvorby disparitní mapy.

## Hloubková mapa

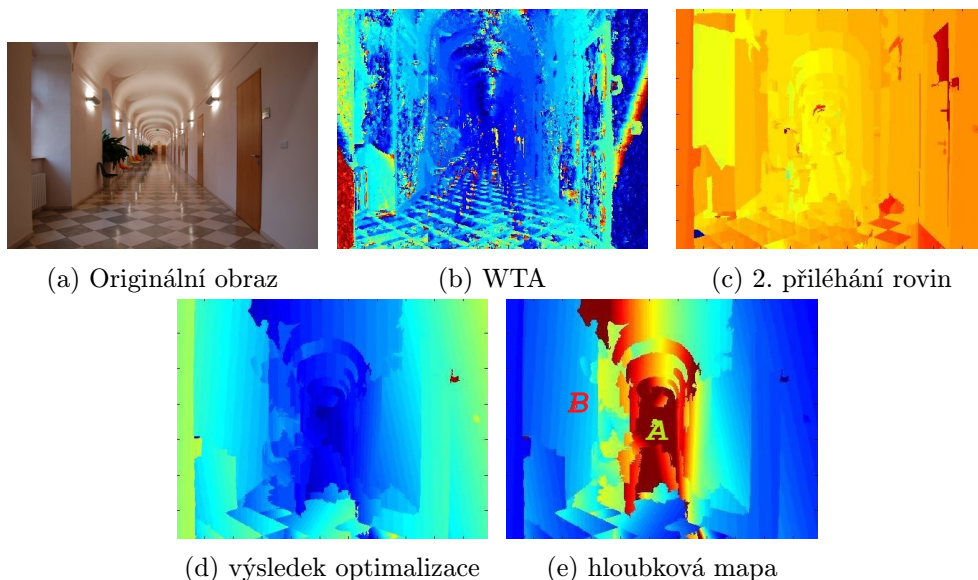
Hloubkové mapy se testovaly na vlastních obrazech. Všechny obrazy byly pořízené na FIT VUT, kde pro každou scénu byly pořízeny 3 obrazy s bází = 40 mm. Byly testovány různé kombinace velikostí a parametrů. Jako výsledná nejlepší kombinace bylo určeno: rozlišení obrazu na  $400 \times 602$  bodů,  $\lambda = 100$ , velikost okna pro metody získávání ceny  $5 \times 5$  bodů. S těmito parametry byly vytvořeny všechny výstupní obrazy.

Před samotným testováním byla učiněna úvaha, jaké bude metoda mít rozlišovací schopnosti. Pro tento účel byla vytvořena tabulka, která nese informace o převodu disparity na hloubku, mimo jiné se zde uvádí velikost chyby, najde-li se disparita  $^+/_-$  jedena úroveň mimo. Protože snímky byly pořizovány všechny se stejným nastavením a všechny jsou upraveny na stejné rozlišení, byla vytvořena jenom jedna tabulka mapující pro názornost velký rozsah vztah mezi disparitou a hloubkou. Je umístěna v příloze C.



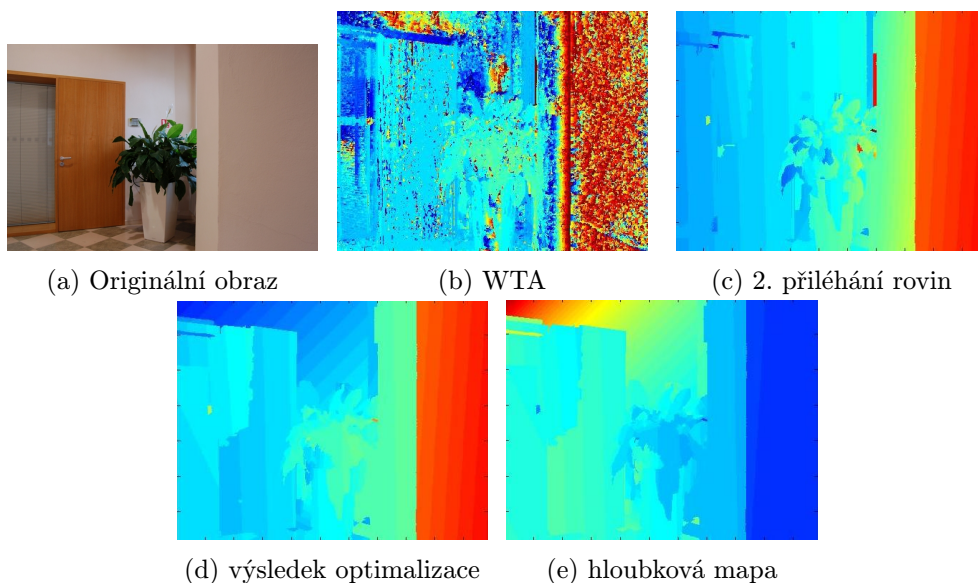
Obrázek 7.4: Schody za mostem

Kolekce obrázků 7.4 obsahuje nejlepší hloubkový snímek, jakého se podařilo dosáhnout. V obraze (e) jsou tři body A,B,C. V těchto bodech byly naměřeny vzdálenosti  $A = 155\text{ cm}$ ,  $B = 248$ ,  $C = 610$  a vypočteny i  $A = 152\text{ cm}$ ,  $B = 232$ ,  $C = 5,77$ . To odpovídá chybovosti pod 5 % .



Obrázek 7.5: Chodba za mostem

Z kolekce 7.5 je poznat, že metoda se potýká s velkými problémy na opakujících se vzorech, kachličkách. Zároveň se jako první ukazuje neschopnost detekovat předměty vzdálené více jak 30 metrů. Rozlišení obrazu je již tak nízké, že tyto změny metoda detekuje již velice špatně. V hloubkové mapě (e) jsou dva body. Bod A je položen na druhý konec chodby, zde vzdálenost byla naměřená 60 metrů, avšak vzdálenost vypočítaná 17 metrů. Bod B je umístěn nad rostlinou v levém výklenku, zde byla naměřená vzdálenost 7,5 metrů, vzdálenost vypočítaná nebyla přesně určena na pozicích, kde by se rostlina měla vyskytovat. Byly naměřeny hodnoty 4.6, 7.54 a 6.12 metrů. Výsledky jsou tedy v této mapě neprokazatelné.



Obrázek 7.6: Rostlina vedle výtahu

Poslední ukázkou hloubkové mapy je obraz s rostlinou 7.6. Tato hloubková mapa je zajímavá tím, že velice dobře vystihuje klady a zápory metody jako celku. V hloubkové mapě



je vidět správná funkčnost algoritmu pro odhad hloubky, správné určení tvarů v obraze. Ale to vše musí být na prvcích scény, které mají strukturu. Jakmile se dostane algoritmus na bílou stěnu, je zcela zmaten a hodnoty jim podávané jsou špatné.

## Diskuze

V testovací části se ukázalo, že algoritmus dokáže věrně rekonstruovat hloubku. To však jen za příznivých okolností, kdy scéna je náhodně strukturovaná. V případech kdy se ve scéně objevilo větší než malé množství stejnobarevných nebo se opakujících oblastí, metoda začala vyvážet chyby. Kdyby bylo těchto „špatných“ oblastí větší množství, metoda by začala chyby zanášet i do dalších částí obrazu. Po určitém počtu iterací se obraz znehodnotil do stavu, kdy lze v obraze vidět základní prvky scény, avšak údaje hloubky vypočítané jsou chaotické a nelze z nich určit jakoukoli hodnotu.

Doba potřebná pro výpočet jedné kolekce se pohybovala od 5 minut do 40 minut. Průměrná doba potřebná pro výpočet je 22 minut a z toho algoritmus stráví 40-50 vteřin výpočtem neoptimalizované disparitní mapy. Zbývajících 21 minut algoritmus provádí optimalizaci. Provádělo se 10 kroků iterace, během kterých vždy algoritmus dospěl do svého minima a cena energie se již dále nesnižovala. Průměrné snížení energie bylo 5-10 %.

## Kapitola 8

### Závěr

Náplní této práce bylo studium a vytváření hloubkových map. V kapitole 2 byly popsány postupy a doprovodné metody získávání disparitních map, kde byl popsán jejich vztah k hloubkové mapě. Kapitola 2 byla zaměřena na studium základních operací pro získání disparitních map a z té následně hloubkové mapy. Kapitola 3 bylo zhodnocení metod výpočtů disparitních map a byla zde vybrána metoda pro implementaci. Kapitola 5 popisuje vybranou metodu. Kapitola 6 uvádí rozdíly při implementaci od předlohy a definuje postupy, které nebyly v kapitole 5 zmíněny. 7. kapitola vyobrazuje výsledky získané při implementaci. A v části diskuze se zmiňuje dosažených výsledků.

Implementace popisována v této práci ukazuje, že měření hloubky z dvou obrazů je možné a lze dosáhnout uspokojivých výsledků, avšak za předpokladu, že scéna má správnou strukturu. Při vytváření disparitních map se často objevovala chyba špatného určování hranic segmentů. Tento problém se vyskytoval i v hloubkových mapách na vlastních snímcích, ale nikoliv v takovém rozsahu. Počet chyb segmentace se stupňoval s klesajícím rozlišením. Lze tedy tomuto problému předejít použitím většího množství dat. Tím však může v důsledku omezené výpočetní kapacity, docházet k nepříjemně dlouhé době výpočtu. Prakticky bylo odzkoušeno, že zpracování obrazu o velikosti více jak  $600 \times 700$  bodů nedospělo ke konci ani po 4 hodinách výpočtů. Metoda byla sice tedy naimplementována dle zadání, její výsledky však mohou být lepší. Zde se nabízejí možnosti pro pokračování projektu: *redukce délky výpočtu*, kdy by se uvažovalo o implementaci v jiném jazyce. *Přenášení chyb v obraze*, problém s přílišnou stejnorodostí scény a problémům spojeným s hledáním korespondencí. *Vyhýbání se lokálním minimům*, optimalizace BP v této implementaci vždy narazili na své minimum do 10. iterace. Krokem zlepšení je najít způsob vyhýbání se jim. Ale i samotná optimalizace metody v Matlabu zajistí by přinesla nemalé úspory ve výpočtu. Při implementaci nebylo zcela vždy využito všech dobrých vlastností Matlabu a někdy pro snadnější pochopení byl zvolen iterativní přístup nad maticovými operacemi. To vše se nabízí jako další možnosti pro pokračování projektu, které mohou metodu značně zkvalitnit.

# Literatura

- [1] Bleyer, M.; Gelautz, M.: Graph-based surface reconstruction from stereo pairs using image segmentation. In *Electronic Imaging 2005*, International Society for Optics and Photonics, 2005, s. 288–299.
- [2] Chris, M. Christoudias and Bogdan Georgescu: Code for the Edge Detection and Image SegmentatiON system.  
<http://coewww.rutgers.edu/riul/research/code/EDISON/index.html>, 2009-5-5 [cit. 2013-7-5].
- [3] Comaniciu, D.; Meer, P.: Mean shift: A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, ročník 24, č. 5, 2002: s. 603–619.
- [4] Daniel, Scharstein and Richard, Szeliski: Middlebury Stereo Vision Page.  
<http://vision.middlebury.edu/stereo/>, 2012-4-5 [cit. 2013-9-5].
- [5] Grgiâc, M.; Mrak, M.; Kunt, M.: *Signals and Communication Technology: High-Quality Visual Experience: Creation, Processing and Interactivity of High-Resolution and High-Dimensional Video Signals*, kapitola The Use of Color Information in Stereo Vision Processing. Springer Berlin Heidelberg, 2010, ISBN 9783642128028, str. 322.
- [6] Klaus, A.; Sormann, M.; Karner, K.: Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, ročník 3, IEEE, 2006, s. 15–18.
- [7] Koch, R.; Huang, F.: *Computer Vision – ACCV 2010 Workshops: ACCV 2010 International Workshops*, kapitola Illumination Invariant Cost Function in Semi-Global Matchin. Springer, 2011, ISBN 9783642228186, s. 246–248.
- [8] Kolmogorov, V.; Zabini, R.: What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, ročník 26, č. 2, 2004: s. 147–159.
- [9] Scharstein, D.; Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, ročník 47, č. 1-3, 2002: s. 7–42.
- [10] Shai Bagon: Shai Bagon's Matlab Code.  
<http://www.wisdom.weizmann.ac.il/~bagon/matlab.html>, 2012-10-29 [cit. 2013-7-5].



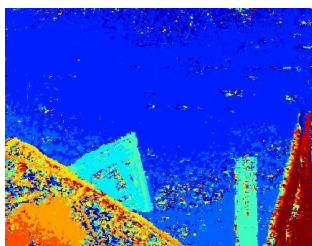
- [11] Shapiro, L.; Stockman, G.: *Computer vision*, kapitola Image Segmentation. Prentice Hall, 2001, ISBN 9780130307965, str. 305.
- [12] Shi, C.; Wang, G.; Pei, X.; aj.: High-accuracy stereo matching based on adaptive ground control points. 2012.
- [13] Zach, C.; Sormann, M.; Karner, K.: Scanline Optimization for Stereo on Graphics Hardware. In *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, 2006, s. 512–518, doi:10.1109/3DPVT.2006.124.

## Příloha A

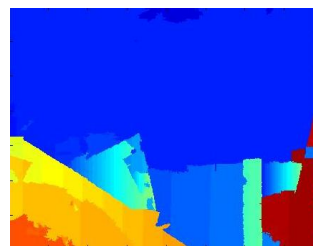
# Disparitní mapy



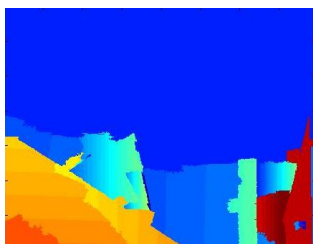
(a) Originální obraz



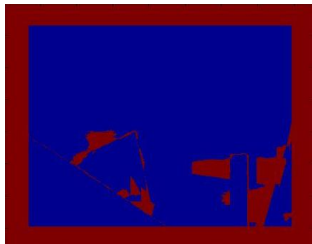
(b) WTA



(c) 2. přiléhání rovin



(d) výsledek optimalizace



(e) chyba

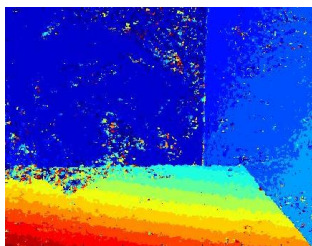


(f) Ground truth disparity

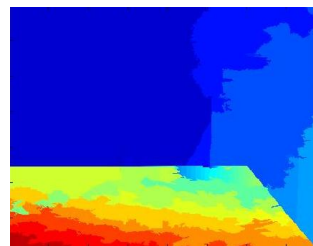
Obrázek A.1: Barn2



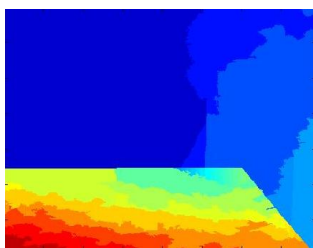
(a) Originální obraz



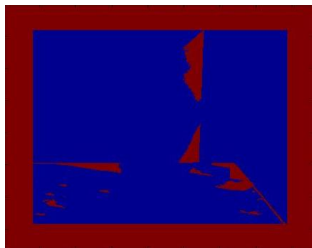
(b) WTA



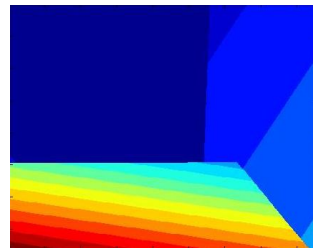
(c) 2. přiléhání rovin



(d) výsledek optimalizace



(e) chyba

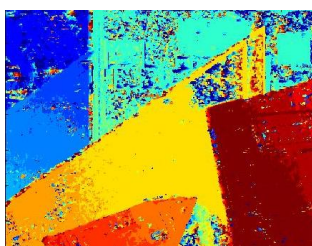


(f) Ground truth disparity

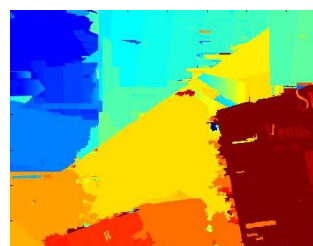
Obrázek A.2: Bull



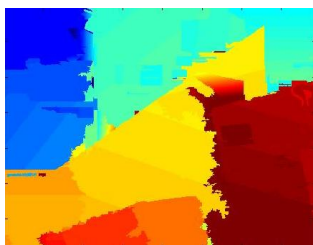
(a) Originální obraz



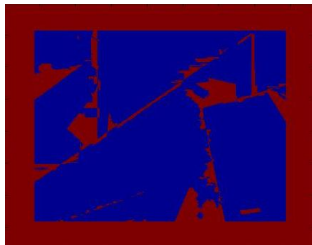
(b) WTA



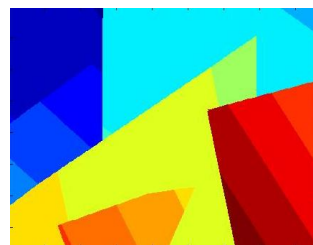
(c) 2. přiléhání rovin



(d) výsledek optimalizace



(e) chyba

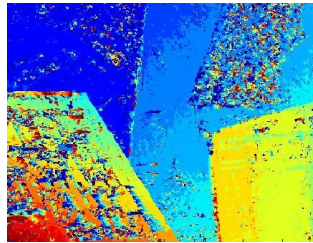


(f) Ground truth disparity

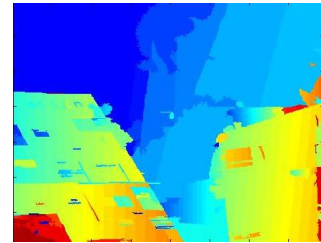
Obrázek A.3: Poster



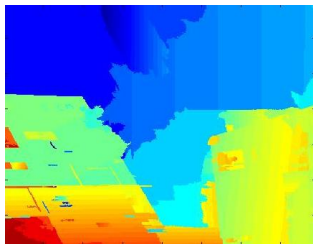
(a) Originální obraz



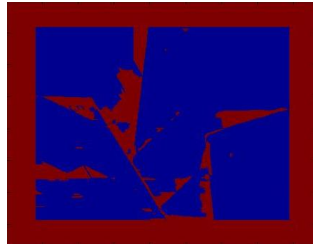
(b) WTA



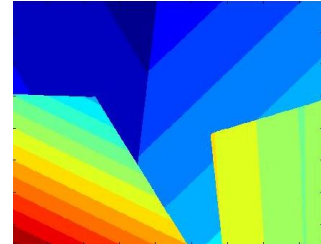
(c) 2. přiléhání rovin



(d) výsledek optimalizace



(e) chyba

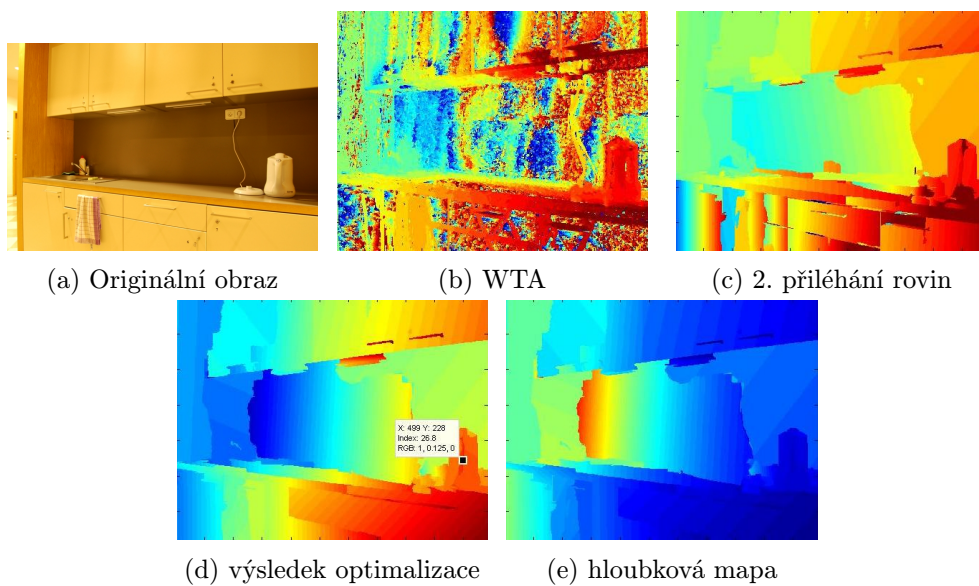


(f) Ground truth disparity

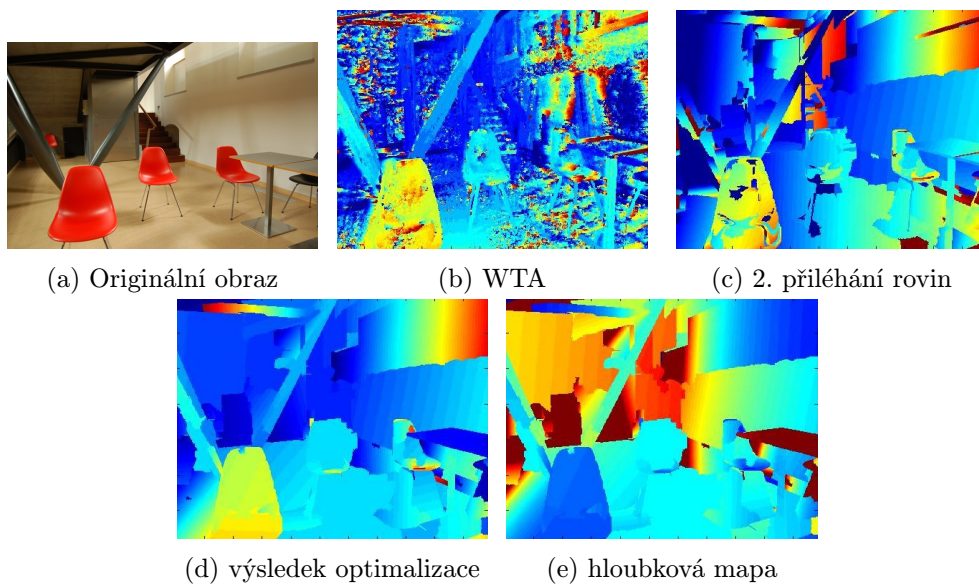
Obrázek A.4: Venus

## Příloha B

# Hloubkové mapy



Obrázek B.1: Kuchyňka



Obrázek B.2: Pod mostem

## Příloha C

### Tabulka přesnosti měření

Tabulka C.1: tabulka přesností  $f_b = 457, 215$   
 $B = 80$

disparita	hloubka	chyba $+/- 1^\circ$	disparita	hloubka	chyba $+/- 1^\circ$
0,00	$\infty$	$\infty$	4,10	8,92	43,50
0,10	365,77	$\infty$	4,20	8,71	41,50
0,20	182,89	24384,80	4,30	8,51	39,60
0,30	121,92	9144,30	4,40	8,31	37,80
0,40	91,44	4877,00	4,50	8,13	36,10
0,50	73,15	3048,10	4,60	7,95	34,60
0,60	60,96	2090,10	4,70	7,78	33,10
0,70	52,25	1524,10	4,80	7,62	31,80
0,80	45,72	1161,20	4,90	7,46	30,50
0,90	40,64	914,40	5,00	7,32	29,30
1,00	36,58	738,90	5,10	7,17	28,10
1,10	33,25	609,60	5,20	7,03	27,10
1,20	30,48	511,60	5,30	6,90	26,10
1,30	28,14	435,40	5,40	6,77	25,10
1,40	26,13	375,20	5,50	6,65	24,20
1,50	24,38	326,60	5,60	6,53	23,30
1,60	22,86	286,90	5,70	6,42	22,50
1,70	21,52	254,00	5,80	6,31	21,80
1,80	20,32	226,50	5,90	6,20	21,00
1,90	19,25	203,20	6,00	6,10	20,30
2,00	18,29	183,30	6,10	6,00	19,70
2,10	17,42	166,30	6,20	5,90	19,00
2,20	16,63	151,50	6,30	5,81	18,40
2,30	15,90	138,60	6,40	5,72	17,90
2,40	15,24	127,20	6,50	5,63	17,30
2,50	14,63	117,20	6,60	5,54	16,80
2,60	14,07	108,40	6,70	5,46	16,30
2,70	13,55	100,50	6,80	5,38	15,80
2,80	13,06	93,40	6,90	5,30	15,40
2,90	12,61	87,10	7,00	5,23	14,90
3,00	12,19	81,40	7,10	5,15	14,50
3,10	11,80	76,20	7,20	5,08	14,10
3,20	11,43	71,50	7,30	5,01	13,70
3,30	11,08	67,20	7,40	4,94	13,40
3,40	10,76	63,30	7,50	4,88	13,00
3,50	10,45	59,80	7,60	4,81	12,70
3,60	10,16	56,50	7,70	4,75	12,30
3,70	9,89	53,50	7,80	4,69	12,00
3,80	9,63	50,70	7,90	4,63	11,70
3,90	9,38	48,10	8,00	4,57	11,40
4,00	9,14	45,80	8,50	4,30	10,80
			9,00	4,06	10,30
			9,50	3,85	10,00



disparita	hloubka	chyba $\pm 1^\circ$	disparita	hloubka	chyba $\pm 1^\circ$
10,00	3,66	36,70	31,50	1,16	3,70
10,50	3,48	33,30	32,00	1,14	3,60
11,00	3,33	30,30	32,50	1,13	3,50
11,50	3,18	27,70	33,00	1,11	3,40
12,00	3,05	25,40	33,50	1,09	3,30
12,50	2,93	23,40	34,00	1,08	3,20
13,00	2,81	21,70	34,50	1,06	3,10
13,50	2,71	20,10	35,00	1,05	3,00
14,00	2,61	18,70	35,50	1,03	2,90
14,50	2,52	17,40	36,00	1,02	2,80
15,00	2,44	16,30	36,50	1,00	2,70
15,50	2,36	15,20	37,00	0,99	2,70
16,00	2,29	14,30	37,50	0,98	2,60
16,50	2,22	13,40	38,00	0,96	2,50
17,00	2,15	12,70	38,50	0,95	2,50
17,50	2,09	12,00	39,00	0,94	2,40
18,00	2,03	11,30	39,50	0,93	2,30
18,50	1,98	10,70	40,00	0,91	2,30
19,00	1,93	10,10	40,50	0,90	2,20
19,50	1,88	9,60	41,00	0,89	2,20
20,00	1,83	9,20	41,50	0,88	2,10
20,50	1,78	8,70	42,00	0,87	2,10
21,00	1,74	8,30	42,50	0,86	2,00
21,50	1,70	7,90	43,00	0,85	2,00
22,00	1,66	7,60	43,50	0,84	1,90
22,50	1,63	7,20	44,00	0,83	1,90
23,00	1,59	6,90	44,50	0,82	1,80
23,50	1,56	6,60	45,00	0,81	1,80
24,00	1,52	6,40	45,50	0,80	1,80
24,50	1,49	6,10	46,00	0,80	1,70
25,00	1,46	5,90	46,50	0,79	1,70
25,50	1,43	5,60	47,00	0,78	1,70
26,00	1,41	5,40	47,50	0,77	1,60
26,50	1,38	5,20	48,00	0,76	3,20
27,00	1,35	5,00	49,00	0,75	3,00
27,50	1,33	4,80	50,00	0,73	2,90
28,00	1,31	4,70	51,00	0,72	2,80
28,50	1,28	4,50	52,00	0,70	2,70
29,00	1,26	4,40	53,00	0,69	2,60
29,50	1,24	4,20	54,00	0,68	2,50
30,00	1,22	4,10	55,00	0,67	2,40
30,50	1,20	3,90	56,00	0,65	2,30
31,00	1,18	3,80	57,00	0,64	2,30

disparita	hloubka	chyba $\pm 1^\circ$	disparita	hloubka	chyba $\pm 1^\circ$
58,00	0,63	2,20	101,00	0,36	0,70
59,00	0,62	2,10	102,00	0,36	0,70
60,00	0,61	2,00	103,00	0,36	0,70
61,00	0,60	2,00	104,00	0,35	0,70
62,00	0,59	1,90	105,00	0,35	0,70
63,00	0,58	1,80	106,00	0,35	0,70
64,00	0,57	1,80	107,00	0,34	0,60
65,00	0,56	1,70	108,00	0,34	0,60
66,00	0,55	1,70	109,00	0,34	0,60
67,00	0,55	1,60	110,00	0,33	0,60
68,00	0,54	1,60	111,00	0,33	0,60
69,00	0,53	1,50	112,00	0,33	0,60
70,00	0,52	1,50	113,00	0,32	0,60
71,00	0,52	1,50	114,00	0,32	0,60
72,00	0,51	1,40	115,00	0,32	0,60
73,00	0,50	1,40	116,00	0,32	0,50
74,00	0,49	1,30	117,00	0,31	0,50
75,00	0,49	1,30	118,00	0,31	0,50
76,00	0,48	1,30	119,00	0,31	0,50
77,00	0,48	1,20	120,00	0,30	0,50
78,00	0,47	1,20	121,00	0,30	0,50
79,00	0,46	1,20	122,00	0,30	0,50
80,00	0,46	1,10	123,00	0,30	0,50
81,00	0,45	1,10	124,00	0,29	0,50
82,00	0,45	1,10	125,00	0,29	0,50
83,00	0,44	1,10	126,00	0,29	0,50
84,00	0,44	1,00	127,00	0,29	0,50
85,00	0,43	1,00			
86,00	0,43	1,00			
87,00	0,42	1,00			
88,00	0,42	0,90			
89,00	0,41	0,90			
90,00	0,41	0,90			
91,00	0,40	0,90			
92,00	0,40	0,90			
93,00	0,39	0,80			
94,00	0,39	0,80			
95,00	0,39	0,80			
96,00	0,38	0,80			
97,00	0,38	0,80			
98,00	0,37	0,80			
99,00	0,37	0,70			
100,00	0,37	0,70			